



.....

```

000000 UU UU TTTTTTTTTT TTTTTTTTTT XX XX TTTTTTTTTT
000000 UU UU TTTTTTTTTT TTTTTTTTTT XX XX TTTTTTTTTT
00 UU UU UU TT TT XX XX TT
00 UU UU UU TT TT XX XX TT
00 UU UU UU TT TT XX XX TT
00 UU UU UU TT TT XX XX TT
00 UU UU UU TT TT XX XX TT
00 UU UU UU TT TT XX XX TT
00 UU UU UU TT TT XX XX TT
00 UU UU UU TT TT XX XX TT
00 UU UU UU TT TT XX XX TT
000000 UUUUUUUUUU TT TT XX XX TT
000000 UUUUUUUUUU TT TT XX XX TT

```

```

LL          IIIIII          SSSSSSSS
LL          IIIIII          SSSSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SSSSSS
LL          II             SSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LLLLLLLLLLLL IIIIII          SSSSSSSS
LLLLLLLLLLLL IIIIII          SSSSSSSS

```



```
0001 0 MODULE OUTTXT (
0002 0 IDENT = 'V04-000'
P 0003 0 %BLISS32[
P 0004 0 ADDRESSING_MODE(EXTERNAL=LONG_RELATIVE,NONEXTERNAL=LONG_RELATIVE)
0005 0 ]
0006 0 ) =
0007 1 BEGIN
0008 1
0009 1 *****
0010 1 *
0011 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0012 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0013 1 * ALL RIGHTS RESERVED.
0014 1 *
0015 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0016 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0017 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0018 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0019 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0020 1 * TRANSFERRED.
0021 1 *
0022 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0023 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0024 1 * CORPORATION.
0025 1 *
0026 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0027 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0028 1 *
0029 1 *
0030 1 *****
0031 1
0032 1 ++
0033 1 FACILITY: DSR (Digital Standard RUNOFF) / DSRPLUS
0034 1
0035 1 ABSTRACT: Outputs a single line of text optionally centered.
0036 1
0037 1
0038 1 ENVIRONMENT: Transportable
0039 1
0040 1 AUTHOR: R.W.Friday CREATION DATE: April, 1979
0041 1
```

OUTTXT  
V04-000

Revision History

D 14  
16-Sep-1984 01:23:16  
14-Sep-1984 13:07:34

VAX-11 Bliss-32 V4.0-742  
[RUNOFF.SRC]OUTTXT.BLI;1

Page 2  
(2)

:	43	0042	1	%SBTTL 'Revision History'
:	44	0043	1	
:	45	0044	1	MODIFIED BY:
:	46	0045	1	
:	47	0046	1	007 KFA00007 Ken Alden 16-Mar-1983
:	48	0047	1	SCA is now initialized the same for both DSR and PLUS
:	49	0048	1	
:	50	0049	1	006 KAD00006 Keith Dawson 07-Mar-1983
:	51	0050	1	Global edit of all modules. Updated module names, idents,
:	52	0051	1	copyright dates. Changed require files to BLISS library.
:	53	0052	1	
:	54	0053	1	--



## Module Level Declarations

```
: 56      0054 1 %SBTTL 'Module Level Declarations'
: 57      0055 1
: 58      0056 1
: 59      0057 1 | TABLE OF CONTENTS:
: 60      0058 1 |
: 61      0059 1 |
: 62      0060 1 | INCLUDE FILES:
: 63      0061 1 |
: 64      0062 1 |
: 65      0063 1 LIBRARY 'NXPORT:XPORT';      ! XPORT Library
: 66      0064 1 REQUIRE 'REQ:RNODEF';      ! RUNOFF variant definitions
: 67      0195 1
: 68      U 0196 1 %IF DSRPLUS %THEN
: 69      U 0197 1 LIBRARY 'REQ:DPLLIB';      ! DSRPLUS BLISS Library
: 70      0198 1 %ELSE
: 71      0199 1 LIBRARY 'REQ:DSRLIB';      ! DSR BLISS Library
: 72      0200 1 %FI
: 73      0201 1
: 74      0202 1 |
: 75      0203 1 | MACROS:
: 76      0204 1 |
: 77      0205 1 |
: 78      0206 1 | EQUATED SYMBOLS:
: 79      0207 1 |
: 80      0208 1 EXTERNAL LITERAL
: 81      0209 1     RINTES : UNSIGNED(8);
: 82      0210 1
: 83      0211 1 |
: 84      0212 1 | OWN STORAGE:
: 85      0213 1 |
: 86      0214 1 OWN
: 87      0215 1     PP_SCA : $H_R_SCA_BLOCK;      !Used in PUSH_SCA, POP_SCA macros (defined in SCA.REQ).
: 88      0216 1 |
: 89      0217 1 | EXTERNAL REFERENCES:
: 90      0218 1 |
: 91      0219 1 |
: 92      0220 1 EXTERNAL
: 93      0221 1     MRA : REF FIXED STRING,
: 94      0222 1     SCA : SCA_DEFINITION,
: 95      0223 1     TSF : TSF_DEFINITION;
: 96      0224 1
: 97      0225 1 EXTERNAL ROUTINE
: 98      0226 1     ENDCHR,
: 99      0227 1     OUTNJ;
```

```
.. 101 0228 1 GLOBAL ROUTINE OUTTXT (TEXT_PTR, TEXT_LENGTH, CENTERED) : NOVALUE =
.. 102 0229 1
.. 103 0230 1 ++
.. 104 0231 1 FUNCTIONAL DESCRIPTION:
.. 105 0232 1
.. 106 0233 1     Outputs a generated line of text in the "official" way.
.. 107 0234 1     This means that an MRA/TSF pair have to be allocated and set up.
.. 108 0235 1     This routine is needed when there is no convenient MRA/TSF pair
.. 109 0236 1     around that can be used for outputting text.
.. 110 0237 1
.. 111 0238 1 FORMAL PARAMETERS:
.. 112 0239 1
.. 113 0240 1     TEXT_PTR is a CH$PTR to the text to be centered and output.
.. 114 0241 1     TEXT_LENGTH is the length of the text.
.. 115 0242 1     CENTERED is the page width in which the text should be centered.
.. 116 0243 1
.. 117 0244 1 IMPLICIT INPUTS:      None
.. 118 0245 1
.. 119 0246 1 IMPLICIT OUTPUTS:     None
.. 120 0247 1
.. 121 0248 1 ROUTINE VALUE:
.. 122 0249 1 COMPLETION CODES:      None
.. 123 0250 1
.. 124 0251 1 SIDE EFFECTS: None
.. 125 0252 1
.. 126 0253 1 --
.. 127 0254 1
.. 128 0255 2 BEGIN
.. 129 0256 2 LOCAL
.. 130 0257 2     HOLD_MRA,
.. 131 0258 2     HOLD_SCA : VECTOR[SCA_SIZE],
.. 132 0259 2     HOLD_TSF,
.. 133 0260 2     TEMP_MRA : FIXED STRING[100],
.. 134 0261 2     TEMP_TSF : VECTOR[TSF_SIZE],
.. 135 0262 2     PTR;
.. 136 0263 2
.. 137 0264 2 !Remember location of current TSF
.. 138 0265 2 HOLD_TSF = .TSF;
.. 139 0266 2 !Set up substitute TSF
.. 140 0267 2 INCR I FROM 0 TO TSF_SIZE - 1 DO TEMP_TSF[I] = 0;
.. 141 0268 2 TSF = TEMP_TSF;
.. 142 0269 2 !Set change-bar character to space.
.. 143 0270 2 TSF_BAR_CHAR = %C' ';
.. 144 0271 2
.. 145 0272 2 !Remember location of current MRA
.. 146 0273 2 HOLD_MRA = .MRA;
.. 147 0274 2 !Set up substitute MRA;
.. 148 0275 2 FS_MAXSIZE(TEMP_MRA) = 100;
.. 149 0276 2 FS_INIT(TEMP_MRA);
.. 150 0277 2 MRA = TEMP_MRA;
.. 151 0278 2
.. 152 0279 2 !Copy SCA before setting up a new one.
.. 153 0280 2 PUSH_SCA; !Save the SAVED SCA bits.
.. 154 0281 2
.. 155 0282 2 INCR I FROM 0 TO SCA_SIZE -1 DO HOLD_SCA[I] = .SCA[I];
.. 156 0283 2
.. 157 0284 2 INCR I FROM 0 TO SCA_SAVE_START -1 DO SCA[I] = 0;
```



## Module Level Declarations

```

: 158      0285 2      INCR I FROM SCA_SAVE_END +1 TO SCA_SIZE -1 DO SCA[I] = 0;
: 159      0286 2      SCA_CC_OK = 0;
: 160      0287 2      SCA_KER = 0;
: 161      0288 2      SCA_AUTOTITLE = 0;
: 162      0289 2
: 163      0290 2      !Initialize SCA.
: 164      0291 2      SCA_FC_CASE = TRUE;
: 165      0292 2      SCA_RM = 150;
: 166      0293 2      SCA_LM = 0;
: 167      0294 2      SCA_SPACING = 1;
: 168      0295 2      SCA_FC = TRUE;
: 169      0296 2      SCA_FILL = TRUE;
: 170      0297 2      SCA_JUSTIFY = TRUE;
: 171      0298 2      SCA_CROCK = TRUE;
: 172      0299 2      SCA_WRD_PNTR = .FS START (MRA);
: 173      0300 2      SCA_WRD_CPEND = RINTES;
: 174      0301 2      !Set change-bar character(s) to space.
: 175      0302 2      SCA_BAR_CHAR = %C' ';
: 176      0303 2      SCA_WRD_BAR_CHR = %C' ';
: 177      0304 2
: 178      0305 2      PTR = .TEXT_PTR;
: 179      0306 2
: 180      0307 2      INCR I FROM 1 TO .TEXT_LENGTH DO
: 181      0308 2          ENDCHR (CH$RCHAR_A(PTR));
: 182      0309 2
: 183      0310 2      !Center the text
: 184      0311 2      TSF_ADJUST = MAX ((.CENTERED - .TEXT_LENGTH)/2, 0);
: 185      0312 2      !Output the line containing the centered text.
: 186      0313 2      OUTNJ ();
: 187      0314 2
: 188      0315 2      !Restore original SCA
: 189      0316 2      INCR I FROM 0 TO SCA_SIZE - 1 DO SCA[I] = .HOLD_SCA[I];
: 190      0317 2
: 191      0318 2      POP_SCA;      !Restore the SAVED SCA bits.
: 192      0319 2
: 193      0320 2      !Restore original MRA
: 194      0321 2      MRA = .HOLD_MRA;
: 195      0322 2
: 196      0323 2      !Restore original TSF
: 197      0324 2      TSF = .HOLD_TSF;
: 198      0325 2
: 199      0326 1      END;
```

!End of OUTTXT

```

                                .TITLE  OUTTXT
                                .IDENT   \V04-000\
                                .PSECT   $OWNS$,NOEXE,2
                                00000 PP_SCA: .BLKB  48
                                .EXTRN   RINTES, MRA, SCA
                                .EXTRN   TSF, ENDCHR, OUTNJ
                                .PSECT   $CODE$,NOWRT,2
                                03FC 00000 .ENTRY  OUTTXT, Save R2,R3,R4,R5,R6,R7,R8,R9      : 0228
```

	59	00000000G	EF	9E	00002	MOVAB	MRA, R9		
	58	00000000G	EF	9E	00009	MOVAB	TSF, R8		
	57	00000000G	EF	9E	00010	MOVAB	PP_SCA, R7		
	56	00000000G	EF	9E	00017	MOVAB	SCA, R6		
	5E	FD6C	CE	9E	0001E	MOVAB	-660(SP), SP		
	54		68	D0	00023	MOVL	TSF, HOLD_TSF		0265
			50	D4	00026	CLRL	I		0267
			6E40	D4	00028	CLRL	TEMP_TSF[I]		
F9	50		27	F3	0002B	AOBLEQ	#39, I, 1\$		
	68		6E	9E	0002F	MOVAB	TEMP_TSF, TSF		0268
	50		68	D0	00032	MOVL	TSF, R0		
1C	A0		20	D0	00035	MOVL	#32, 28(R0)		0270
	55		69	D0	00039	MOVL	MRA, HOLD_MRA		0273
00A8	CE	64	8F	9A	0003C	MOVZBL	#100, TEMP_MRA+8		0275
		00AC	CE	D4	00042	CLRL	TEMP_MRA+12		0276
00A0	CE	00B0	CE	9E	00046	MOVAB	TEMP_MRA+16, TEMP_MRA		
00A4	CE	00A0	CE	D0	0004D	MOVL	TEMP_MRA, TEMP_MRA+4		
	69	00A0	CE	9E	00054	MOVAB	TEMP_MRA, MRA		0277
	67	64	B6	D0	00059	MOVL	@SCA+100, PP_SCA		
04	A7	68	B6	D0	0005D	MOVL	@SCA+104, PP_SCA+4		
08	A7	6C	B6	D0	00062	MOVL	@SCA+108, PP_SCA+8		
0C	A7	70	B6	D0	00067	MOVL	@SCA+112, PP_SCA+12		
10	A7	74	B6	D0	0006C	MOVL	@SCA+116, PP_SCA+16		
14	A7	78	B6	D0	00071	MOVL	@SCA+120, PP_SCA+20		
18	A7	7C	B6	D0	00076	MOVL	@SCA+124, PP_SCA+24		
1C	A7	0080	D6	D0	0007B	MOVL	@SCA+128, PP_SCA+28		
20	A7	0084	D6	D0	00081	MOVL	@SCA+132, PP_SCA+32		
24	A7	0088	D6	D0	00087	MOVL	@SCA+136, PP_SCA+36		
28	A7	008C	D6	D0	0008D	MOVL	@SCA+140, PP_SCA+40		
2C	A7	0090	D6	D0	00093	MOVL	@SCA+144, PP_SCA+44		
			50	D4	00099	CLRL	I		0282
F1	0114	CE40	6640	D0	0009B	MOVL	SCA[I], HOLD_SCA[I]		
	50	0000005F	8F	F3	000A2	AOBLEQ	#95, I, 2\$		0284
			50	D4	000AA	CLRL	I		
			6640	D4	000AC	CLRL	SCA[I]		
F9	50		18	F3	000AF	AOBLEQ	#24, I, 3\$		
	50		25	D0	000B3	MOVL	#37, I		0285
			6640	D4	000B6	CLRL	SCA[I]		
F5	50	0000005F	8F	F3	000B9	AOBLEQ	#95, I, 4\$		
		6C	B6	D4	000C1	CLRL	@SCA+108		0286
		0084	D6	D4	000C4	CLRL	@SCA+132		0287
		008C	D6	D4	000C8	CLRL	@SCA+140		0288
00D0	C6		01	D0	000CC	MOVL	#1, SCA+208		0291
78	B6	96	8F	9A	000D1	MOVZBL	#150, @SCA+120		0292
		74	B6	D4	000D6	CLRL	@SCA+116		0293
7C	B6		01	D0	000D9	MOVL	#1, @SCA+124		0294
0094	C6		01	D0	000DD	MOVL	#1, SCA+148		0295
68	B6		01	D0	000E2	MOVL	#1, @SCA+104		0296
64	B6		01	D0	000E6	MOVL	#1, @SCA+100		0297
70	B6		01	D0	000EA	MOVL	#1, @SCA+112		0298
00F8	C6	00	B9	D0	000EE	MOVL	@MRA, SCA+248		0299
0118	C6	00G	8F	9A	000F4	MOVZBL	#RINTES, SCA+280		0300
0088	D6		20	D0	000FA	MOVL	#32, @SCA+136		0302
0114	C6		20	D0	000FF	MOVL	#32, SCA+276		0303
	53	04	AC	D0	00104	MOVL	TEXT_PTR, PTR		0305
			52	D4	00108	CLRL	I		0307
			0A	11	0010A	BRB	6\$		



		7E		83	9A	0010C	5\$:	MOVZBL	(PTR)+, -(SP)		0308
		EF		01	FB	0010F		CALLS	#1, ENDCHR		
F1	00000000G	52	08	AC	F3	00116	6\$:	AOBLEQ	TEXT_LENGTH, 1, 5\$		
		51		68	D0	0011B		MOVL	TSF, R1		
50	0C	AC	08	AC	C3	0011E		SUBL3	TEXT_LENGTH, CENTERED, R0		0311
		50		02	C6	00124		DIVL2	#2, R0		
				02	18	00127		BGEQ	7\$		
	28	A1		50	D4	00129		CLRL	R0		
	00000000G	EF		50	D0	0012B	7\$:	MOVL	R0, 40(R1)		
				00	FB	0012F		CALLS	#0, OUTNJ		0313
				50	D4	00136		CLRL	I		0316
F1	6640	0114	CE40	D0	00138	8\$:	MOVL	HOLD_SCA[I], SCA[I]			
	50	0000005F	8F	F3	0013F		AOBLEQ	#95, I, 8\$			
	64	B6		67	D0	00147		MOVL	PP_SCA, @SCA+100		
	68	B6	04	A7	D0	0014B		MOVL	PP_SCA+4, @SCA+104		
	6C	B6	08	A7	D0	00150		MOVL	PP_SCA+8, @SCA+108		
	70	B6	0C	A7	D0	00155		MOVL	PP_SCA+12, @SCA+112		
	74	B6	10	A7	D0	0015A		MOVL	PP_SCA+16, @SCA+116		
	78	B6	14	A7	D0	0015F		MOVL	PP_SCA+20, @SCA+120		
	7C	B6	18	A7	D0	00164		MOVL	PP_SCA+24, @SCA+124		
0080	D6		1C	A7	D0	00169		MOVL	PP_SCA+28, @SCA+128		
0084	D6		20	A7	D0	0016F		MOVL	PP_SCA+32, @SCA+132		
0088	D6		24	A7	D0	00175		MOVL	PP_SCA+36, @SCA+136		
008C	D6		28	A7	D0	0017B		MOVL	PP_SCA+40, @SCA+140		
0090	D6		2C	A7	D0	00181		MOVL	PP_SCA+44, @SCA+144		
	69			55	D0	00187		MOVL	HOLD_MRA, MRA		0321
	68			54	D0	0018A		MOVL	HOLD_TSF, TSF		0324
				04	0018D		RET				0326

; Routine Size: 398 bytes, Routine Base: \$CODE\$ + 0000

: 200 0327 1  
: 201 0328 1 END  
: 202 0329 0 ELUDOM

!End of module

## PSECT SUMMARY

Name	Bytes	Attributes
\$OWNS	48 NOVEC, WRT, RD	, NOEXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)
\$CODE\$	398 NOVEC, NOWRT, RD	, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

## Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]XPORT.L32;1	590	0	0	252	00:00.1

OUTTXT  
V04-000

Module Level Declarations

J 14  
16-Sep-1984 01:23:16  
14-Sep-1984 13:07:34

VAX-11 Bliss-32 V4.0-742  
[RUNOFF.SRC]OUTTXT.BLI;1

Page 8  
(4)

: \_\$255\$DUA28:[RUNOFF.SRC]DSRLIB.L32;1 1248 53 4 86 00:00.3

COMMAND QUALIFIERS

: BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LISS:OUTTXT/OBJ=OBJ\$:OUTTXT MSRC\$:OUTTXT/UPDATE=(ENH\$:OUTTXT)

: Size: 398 code + 48 data bytes  
: Run Time: 00:11.8  
: Elapsed Time: 00:26.9  
: Lines/CPU Min: 1670  
: Lexemes/CPU-Min: 16406  
: Memory Used: 94 pages  
: Compilation Complete



0346 AH-BT13A-SE  
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION  
CONFIDENTIAL AND PROPRIETARY

